SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Teo Turković

HOMOMORFNA ENKRIPCIJA I ELEKTRONIČKO GLASOVANJE

Diplomski rad

Voditelj rada:
doc. dr. sc. Matija Kazalicki

Zagreb, studeni 2017.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Sadržaj

Uvod

Cilj ovog diplomskog rada je izložiti osnove homomorfne enkripcije s primjenom na elektroničko glasovanje.

Homomorfna enkripcija pruža trećim osobama mogućnost obavljanja nekih jednostavnih izračuna na šifriranim podacima bez otkrivanja bilo kakvih informacija o samim podacima. Treća osoba može izračunati šifriranu sumu dva šifrirana broja ili šifriran produkt dvije šifrirane poruke. To je moguće jer je funkcija šifriranja homomorfizam grupa pa čuva grupne operacije. To čini homomorfnu enkripciju korisnom u raznim protokolima za koje je nužno očuvanje privatnosti.

Povijest homomorfnog šifriranja

Godine 1978. (ubrzo nakon objavljivanja RSA kriptosustava) Ronald Rivest, Leonard Adleman i Michael Dertouzos su prvi puta predložili koncept homomorfne enkripcije. U idućih 30 godina je napravljeno jako malo napretka. Sustav šifriranja Shafija Goldwassera i Silvia Micalia predložen je 1982. godine i to je bio prvi dokazivi sustav enkripcije koji je dosegao izvanrednu razinu sigurnosti, aditivno je homomorfan, ali može šifrirati samo jedan bit. Pascal Paillier je 1999. godine predložio dokaziv sustav za šifriranje koji je također bio aditivno homomorfna enkripcija. Nekoliko godina kasnije, 2005. godine, Dan Boneh, Eu-Jin Goh i Kobi Nissim izumili su sustav dokazivih enkripcija, s kojima je moguć neograničen broj zbrajanja, ali se može izvršiti samo jedno množenje.

Motivacija: Glasovanje na oglasnoj ploči

Htjeli bismo iskoristiti $E(m_1) \cdot E(m_2) = E(m_1 + m_2)$ tako da možemo zbrajati glasove. Zamislimo da imamo veliku javnu oglasnu ploču sa kandidatima A, B i C i glasačima $1, \ldots, n$.

Ako imamo aditivno homomorfno svojstvo, moramo dekriptirati samo brojke u retku ukupnih iznosa, čime bi se osigurala privatnost glasača. Želimo pronaći $E(U_j) = \prod_i g_{i,j}$.

	A	B	C
Glasač 1	1	0	0
Glasač 2	0	1	0
Glasač 3	1	0	0
Ukupno	2	1	0

Tablica 0.1: S redovitom oglasnom pločom imamo glasačke listiće u otvorenom tekstu. Unos 1 predstavlja glas za kandidata.

	A	B	C
Glasač 1	$g_{1,A}$	$g_{1,B}$	$g_{1,C}$
Glasač 2	$g_{2,A}$	$g_{2,B}$	$g_{2,C}$
Glasač 3	$g_{3,A}$	$g_{3,B}$	$g_{3,C}$
Ukupno	g_A	g_B	g_C

Tablica 0.2: $g_{1,A}$ i sl. označavaju glasove kriptirane pomoću nekog kriptosustava s javnim ključem, pri čemu je privatni ključ u vlasništvu izbornih dužnosnika.

Šifriranje mora biti naizgled slučajno; šifrirane 0 moraju izgledati drugačije, tako da napadač neće biti u stanju utvrditi jesu li dvije osobe napravile isti glas jednostavno provjeravajući jesu li šifrirani tekstovi isti. Slično tome, šifrati od 1 također moraju izgledati različito. Dakle, jednostavni RSA kriptosustav nije prikladan za ovu primjenu.

Poglavlje 1

Uvod u kriptografiju

1.1 Osnovni pojmovi

Kriptografija je znanstvena disciplina koja se bavi proučavanjem metoda za slanje poruka u takvom obliku da ih samo onaj kome su namijenjene može pročitati.

Osnovni zadatak kriptografije je omogućiti dvjema osobama, *pošiljaocu* i *primaocu* (u kriptografskoj literaturi *Alice* i *Bob*), komuniciranje preko nesigurnog komunikacijskog kanala (telefonska linija, računalna mreža i sl.) na način da treća osoba, njihov protivnik (u literaturi *Eva* ili *Oskar*) koji može nadzirati komunikacijski kanal, ne može razumjeti njihove poruke. Poruku koju pošiljaoc želi poslati primaocu zvat ćemo *otvoreni tekst*. To može biti tekst na njihovom materinjem jeziku, numerički podaci ili bilo što drugo. Pošiljaoc transformira otvoreni tekst koristeći unaprijed dogovoreni *ključ*. Taj postupak se naziva *šifriranje*, a dobiveni rezultat *šifrat* ili *kriptogram*. Nakon toga pošiljaoc pošalje šifrat preko nekog komunikacijskog kanala. Protivnik prisluškujući može doznati sadržaj šifrata, ali ne može odrediti otvoreni tekst. Za razliku od njega, primaoc koji zna ključ kojim je šifrirana poruka može *dešifrirati* šifrat i odrediti otvoreni tekst.

Za razliku od dešifriranja, *kriptoanaliza* ili *dekriptiranje* je znanstvena disciplina koja se bavi proučavanjem postupaka za čitanje skrivenih poruka bez poznavanja ključa. *Kriptologija* je grana znanosti koja obuhvaća kriptografiju i kriptoanalizu.

Kriptografski algoritam ili *šifra* je matematička funkcija koja se koristi za šifriranje i dešifriranje. Općenito, radi se o dvije funkcije, jednoj za šifriranje, a drugoj za dešifriranje. Te funkcije preslikavaju osnovne elemente otvorenog teksta (najčešće su to slova, bitovi, grupe slova ili bitova) u osnovne elemente šifrata i obratno. Funkcije se biraju iz određene familije funkcija u ovisnosti o ključu. Skup svih mogućih vrijednosti ključeva nazivamo *prostor ključeva*. *Kriptosustav* se sastoji od kriptografskog algoritma te svih mogućih otvorenih tekstova, šifrata i ključeva.

Definicija 1.1.1. *Kriptosustav je uređena petorka* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *za koju vrijedi:*

- $\mathcal{P}$ *je konačan skup svih mogućih osnovnih elemenata otvorenog teksta;*
- $\mathcal{C}$ *je konačan skup svih mogućih osnovnih elemenata šifrata;*
- $\mathcal{K}$ *je prostor ključeva, tj. konačan skup svih mogućih ključeva;*
- *Za svaki* $K \in \mathcal{K}$ *postoji funkcija šifriranja* $e_K \in \mathcal{E}$ *te njoj odgovarajuća funkcija dešifriranja* $d_K \in \mathcal{D}$. *Pritom su* $e_K : \mathcal{P} \to \mathcal{C}$ *i* $d_K : \mathcal{C} \to \mathcal{P}$ *funkcije sa svojstvom da je*

$$d_K(e_K(x)) = x$$

za svaki otvoreni tekst $x \in \mathcal{P}$.

Ako su funkcija šifriranja e_K i funkcija dešifriranja d_K jednake ili se d_K može lako izračunati iz e_K, tada takav kriprosustav nazivamo *simetrični* jer se i šifriranje i dešifriranje može izvesti samo poznavanjem vrijednosti e_K. U simetričnom kriptosustavu vrijednost e_K očito mora ostati skrivena pa takve kriptosustave nazivamo i *kriptosustavi s tajnim ključem*. Ako je računalno neprovedivo poznavajući samo funkciju za šifriranje e_K izračunati funkciju za dešifriranje d_K, tada takav kriptosustav nazivamo *kriptosustav s javnim ključem*, budući da tada funkcija e_K može biti javna. Pod računalno neprovedivo, smatramo da napadač (koji je ograničen računalnom snagom) ima samo beznačajnu vjerojatnost da će uspjeti izračunati d_k poznavajući samo e_K, s obzirom na neki predefinirani sigurnosni parametar ϵ. U mnogim slučajevima, ϵ je duljina javnog ključa u bitovima.

Definicija 1.1.2. *Za funkciju* $\nu : \mathbb{N} \to \mathbb{R}$ *kažemo da je* beznačajna *ako za bilo koji ne-nul polinom* $p \in \mathbb{R}[x]$ *postoji* $m \in \mathbb{N}$ *takav da za svaki* $n > m$ *vrijedi:*

$$|\nu(n)| < \frac{1}{|p(n)|}.$$

U kriptosustavima s javnim ključem sve informacije potrebne za provedbu šifriranja nazivamo *javni ključ*, a sve informacije potrebne za provedbu dešifriranja nazivamo *privatni* ili *tajni ključ*. U simetričnim kriptosustavima sve informacije za računanje e_K (a time i d_K) nazivamo *tajni ključ*. Najpopularniji i najšire korišteni kriptosustav s javnim ključem je *RSA* kriptosustav kojeg su 1977. godine izumili *Rivest*, *Shamir* i *Adleman*.

Definicija 1.1.3. RSA kriptosustav

Neka su n *i* p *prosti brojevi, definirajmo* $n = pq$. *Neka je* $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$ *te*

$$\mathcal{K} = \{(n, p, q, d, e) \mid de \equiv 1 \bmod \phi(n)\}$$

gdje je ϕ Eulerova funkcija. *Za* $K = (n, p, q, d, e)$ *definiramo*

$$e_K(x) = x^e \bmod n$$

i

$$d_K(y) = y^d \bmod n$$

za x, y $\in \mathbb{Z}_n$*. Vrijednosti n i e su javne, a vrijednosti p, q i d su tajne.*

Lako je provjeriti da je zadovoljeno četvrto svojstvo iz Definicije 1.1.1:

$$d_K(e_K(x)) \equiv (e_K(x))^d \equiv (x^e)^d \equiv x^{de} \bmod n$$

Kako je $de \equiv 1 \bmod \phi(n)$, to znači da postoji prirodan broj t takav da je $de = t \cdot \phi(n) + 1$ pa imamo

$$x^{de} = x^{t\cdot\phi(n)+1} = x^{t\cdot\phi(n)} \cdot x = \left[x^{\phi(n)}\right]^t \cdot x$$

U zavisnosti od n i x imamo 2 slučaja:

1. $nzd(x, n) = 1$
   tada po Eulerovom teoremu vrijedi $x\phi(n) \equiv 1 \bmod n$ pa je

   $$x^{de} \equiv 1^t \cdot x \equiv x \bmod n.$$

2. $nzd(x, n) \neq 1$
   Ako je $nzd(x, n) = n$, tada je $x = 0$ pa je kongruencija trivijalno zadovoljena. Neka je $nzd(x, n) = p$ ili $nzd(x, n) = q$. Bez smanjenja općenitosti uzmimo da je $nzd(x, n) = p$, pa je $x^{de} \equiv 0 \equiv x \bmod p$. Kako je $nzd(x, pq) = p$, gdje su p i q prosti, to je $nzd(x, q) = 1$ pa je prema Eulerovom teoremu

   $$x^{\phi(q)} \equiv 1 \bmod q \rightarrow x^{q-1} \equiv 1 \bmod q.$$

   Sada je

   $$x^{de} = \left(x^{q-1}\right)^{(p-1)\cdot t} \cdot x \equiv x \bmod q.$$

   Konačno je $x^{de} \equiv x \bmod pq$, tj. $x^{de} \equiv x \bmod n$.

Preostaje nam vidjeti da napadaču nije dovoljno samo poznavanje funkcije za šifriranje e_K da bi mogao odrediti funkciju za dešifriranje d_K. Pretpostavimo da napadač zna vrijednosti n i e. Da bi mogao dešifrirati poruku pomoću d_K, napadač očito mora saznati vrijednost d, jedinstveni cijeli broj takav da je $de \equiv 1 \bmod \phi(n)$. Ako znamo vrijednost $\phi(n) = \phi(pq) = (p-1)(q-1)$, to je moguće lako izračunati (koristeći Euklidov algoritam), ali budući da su p i q tajni, napadač prvo mora faktorizirati n da odredi p i q. Za dovoljno veliki n i činjenicu da su računalni resursi ograničeni, napadač ima samo beznačajnu vjerojatnost da će faktorizirati n pa time naći $\phi(n)$ i d.

Napadač ne mora uvijek saznati privatni ključ ako i bez te informacije može otkriti sadržaj poruke. Primjetimo da u osnovnom *RSA* kriptosustavu, ako su dvije poruke $m_1 =$

m_2 šifrirane, tada su c_1 i c_2 identični. Drugim riječima, šifriranje je deterministički proces. Kad bi napadač promatrao šifrat c, lako može provjeriti je li c šifrat otvorenog teksta m tako da provjeri vrijedi li $e_K(m) = c$. Napadač bi tako mogao konstruirati preslikavanje mogućih otvorenih tekstova u njihove šifrate i tako vrlo lako otkriti sadržaj mnogih šifriranih poruka. Da bi se zaštitili od takvog napada, često zahtijevamo da šifrate kreirane kriptosustavom bude teško razlikovati. To znači da napadač, ako ima jedan šifrat poznate poruke iz skupa $\{m_0, m_1\}$, ne može odrediti koju poruku šifrat predstavlja sa vjerojatnošću značajno većom od $\frac{1}{2}$. To ćemo formalno definirati u sljedećoj definiciji.

Definicija 1.1.4. *Za kriptosustav kažemo da je nerazlučiv prilikom napada odabranim otvorenim tekstom, ili $IND-CPA$ (eng. Indistinguishable under chosen plaintext attack), ako proizvoljno polinomijalno ograničeni napadač ne može pobijediti u sljedećoj igri sa vjerojatnošću većom od $\frac{1}{2} + \nu(\epsilon)$, gdje je ϵ predefinirani sigurnosni parametar, a funkcija ν je beznačajna:*

1. *Izazivač kreira instancu kriptosustava koristeći sigurnosni parametar ϵ te šalje javni ključ napadaču.*
2. *Napadač može izvesti polinomijalno mnogo šifriranja ili drugih operacija.*
3. *Napadač odabire dvije proizvoljne poruke $m_0, m_1 \in \mathcal{P}$ i šalje ih izazivaču.*
4. *Izazivač slučajnim odabirom bira bit $b \in \{0, 1\}$ i šalje $e_K(m_b)$ napadaču.*
5. *Napadač ponovno smije izvesti polinomijalno mnogo šifriranja ili drugih operacija, zatim odgovara sa* 0 *ili* 1.

Napadač pobjeđuje u igri ako je bit-vrijednost koju je odabrao napadač jednaka vrijednosti koju je odabrao izazivač.

Izraz *polinomijalno siguran* se ponekad koristi umjesto $IND-CPA$. Neki kriptosustavi, zvani *vjerojatnosni kriptosustavi*, su kreirani tako da je slučajnost ugrađena u funkciju šifriranja. U vjerojatnosnom kriptosustavu je funkcija šifriranja funkcija skupa elemenata otvorenog teksta i nekog slučajno odabranog parametra.

Definicija 1.1.5. *Vjerojatnosni kriptosustav je uređena šestorka $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{R})$ gdje su $\mathcal{P}$, $\mathcal{C}$ i $\mathcal{K}$ definirani kao u 1.1.1, a $\mathcal{R}$ je slučajan skup za šifriranje. Kao dodatak uz osnovna svojstva kriptosustava, u vjerojatnosnom kriptosustavu moraju vrijediti i sljedeća dva svojstva:*

- *Svaka funkcija šifriranja $e_K : \mathcal{P} \times \mathcal{R} \to \mathcal{C}$ i pripadna funkcija dešifriranja $d_K : \mathcal{C} \to \mathcal{P}$ su funkcije takve da vrijedi*

$$d_K(e_K(x, r)) = x$$

za svaki otvoreni tekst $x \in \mathcal{P}$ i svaki $r \in \mathcal{R}$.

- *Neka je ϵ zadani sigurnosni parametar. Za bilo koji fiksirani $K \in \mathcal{K}$ i svaki $x \in \mathcal{P}$, definiramo funkciju distribucije vjerojatnosti $p_{K,x}$ na C gdje je $p_{K,x}(y)$ vjerojatnost da je y šifrat uz ključ K za sve moguće vrijednosti $r \in \mathcal{R}$. Neka su $x, x' \in \mathcal{P}$, $x \neq x'$ i $K \in \mathcal{K}$. Tada su sve funkcije distribucije vjerojatnosti $p_{K,x}$ i $p_{K,x'}$ ϵ-nerazlučive u polinomijalnom vremenu ($|\sum_{y \in C} p_{K,x}(y) - \sum_{y \in C} p_{K,x'}(y)| \leq \epsilon$).*

Nerazlučivost prilikom napada odabranim otvorenim tekstom je najčešće najslabiji oblik sigurnosti koju očekujemo od kriptosustava. Uistinu, priroda kriptosustava s javnim ključem dozvoljava napadaču da šifrira polinomijalno mnogo proizvoljnih poruka bez da ima pristup nekom posebnom hardveru ili tajnim informacijama. U mnogim slučajevima možemo pretpostaviti da napadač može izvesti puno kompliciranije radnje od odabira otvorenog teksta i računanja pripadnih šifrata. Možda napadač može odabrati proizvoljan šifrat i vidjeti pripadni otvoreni tekst. Takvu vrstu napada nazivamo *Napad odabranim šifratom*. Da bi formalizirali ovu ideju, pretpostavimo da postoji crna kutija koja prihvaća šifrat kao ulaz, a kao izlaz vraća pripadni otvoreni tekst.

Definicija 1.1.6. *Za kriptosustav kažemo da je nerazlučiv prilikom napada neprilagodljivim odabranim šifratom, ili $IND - CCA1$ siguran (eng. Indistinguishable under non-adaptive chosen chipertext attack), ako proizvoljno polinomijalno ograničeni napadač ne može pobijediti u sljedećoj igri sa vjerojatnošću većom od $\frac{1}{2} + \nu(\epsilon)$, gdje je ϵ predefinirani sigurnosni parametar, a funkcija ν je beznačajna:*

1. *Izazivač kreira instancu kriptosustava koristeći sigurnosni parametar ϵ te šalje javni ključ napadaču.*
2. *Napadač dobiva pristup crnoj kutiji te zatim on može izvesti polinomijalno mnogo šifriranja, dešifriranja ili drugih operacija.*
3. *Napadač odabire dvije proizvoljne poruke $m_0, m_1 \in \mathcal{P}$ i šalje ih izazivaču.*
4. *Izazivač slučajnim odabirom bira bit $b \in \{0, 1\}$ i šalje $e_K(m_b)$ napadaču.*
5. *Napadač više nema pristup crnoj kutiji za dešifriranje te može izvesti polinomijalno mnogo šifriranja ili drugih operacija (osim dešifriranja), zatim odgovara sa* 0 *ili* 1.

Napadač pobjeđuje u igri ako je bit-vrijednost koju je odabrao napadač jednaka vrijednosti koju je odabrao izazivač.

Napad neprilagodljivim odabranim šifratom se ponekad naziva i *lunchtime attack* zato što napadač može ući u zaštićeni sustav dok je vlasnik na ručku, ali se ne može osloniti na pristup sustavu kasnije. Najjači oblik nerazlučivosti pretpostavlja da napadaču ostane pristup sustavu (crna kutija za dešifriranje) i nakon što je dobio šifrat $e_K(m_b)$ od izazivača,

ali crna kutija neće dati odgovor za ulaz $e_K(m_b)$ tako da napadač može dešifrirati bilo što drugo osim izravno traženog šifrata iz igre.

Definicija 1.1.7. *Za kriptosustav kažemo da je nerazlučiv prilikom napada prilagodljivim odabranim šifratom, ili $IND-CCA2$ siguran (eng. Indistinguishable under adaptive chosen chipertext attack), ako proizvoljno polinomijalno ograničeni napadač ne može pobijediti u sljedećoj igri sa vjerojatnošću većom od $\frac{1}{2} + \nu(\epsilon)$, gdje je ϵ predefinirani sigurnosni parametar, a funkcija ν je beznačajna:*

1. *Izazivač kreira instancu kriptosustava koristeći sigurnosni parametar ϵ te šalje javni ključ napadaču.*
2. *Napadač dobiva pristup crnoj kutiji te zatim on može izvesti polinomijalno mnogo šifriranja, dešifriranja ili drugih operacija.*
3. *Napadač odabire dvije proizvoljne poruke $m_0, m_1 \in \mathcal{P}$ i šalje ih izazivaču.*
4. *Izazivač slučajnim odabirom bira bit $b \in \{0, 1\}$ i šalje $c = e_K(m_b)$ napadaču.*
5. *Crna kutija za dešifriranje je izmijenjena tako da neće odgovoriti ako je ulazni šifrat c, ali će za bilo koji drugi valjani šifrat vratiti pripadni otvoreni tekst. Napadač može izvesti polinomijalno mnogo šifriranja, dešifriranja ili drugih operacija te zatim odgovoriti sa* 0 *ili* 1.

Napadač pobjeđuje u igri ako je bit-vrijednost koju je odabrao napadač jednaka vrijednosti koju je odabrao izazivač.

Iz prethodnih definicija je očito da $IND-CCA2$ podrazumijeva $IND-CCA1$ koji podrazumijeva $IND-CPA$.

U napadima odabranim otvorenim tekstom i napadima odabranim šifratom koje smo dosad razmatrali, kriptosustav smo smatrali sigurnim ako napadač ne može razlučiti o šifratu koje poruke se radi. Međutim, u nekim situacijama bi napadač mogao modificirati šifrat tako da deterministički utječe na otvoreni tekst. Kriptosustave u kojima je ovo moguće nazivamo *plastični kriptosustavi* (eng. malleable) dok kriptosustave koji nemaju to svojstvo nazivamo *ne-plastični*.

Definicija 1.1.8. *Za kriptosustav kažemo da je plastičan ako je za dani šifrat c koji predstavlja otvoreni tekst m moguće izračunati funkcije f i g tako da se $f(c)$ može dešifrirati kao $g(m)$. Ako nije moguće naći takve funkcije f i g, tada za kriptosustav kažemo da je ne-plastičan.*

Očito, bilo koji plastičan kriptosustav ne može biti $IND-CCA2$ siguran jer bi napadač jednostavno mogao za dani $c = e_K(m_b)$ crnoj kutiji za dešifriranje za ulaz poslati $f(c)$ te dobiti $g(m_b)$. Napadač tada izračunava $g^{-1}(g(m_b)) = m_b$ kako bi odredio bit $b \in \{0, 1\}$ koji je izazivač odabrao.

Prisjetimo se RSA kriptosustava iz definicije 1.1.3. Za javni ključ $K = (n, e)$ šifrati poruka m_1 i m_2 su izračunati kao $e_K(m_1) = m_1^e \bmod n$ i $e_K(m_2) = m_2^e \bmod n$. Računajući produkt ta dva šifrata dobivamo

$$\begin{aligned} e_K(m_1)e_K(m_2) \bmod n &= m_1^e m_2^e \bmod n \\ &= (m_1 m_2)^e \bmod n \\ &= e_K(m_1 m_2) \end{aligned}$$

pa je RSA kriptosustav plastičan, budući da možemo izračunati šifrat umnoška otvorenih tekstova samo poznavajući šifrate i javni dio ključa. Stoga RSA kriptosustav nije $IND-CCA2$ siguran.

Poglavlje 2

Homomorfni kriptosustavi

Homomorfna enkripcija je oblik šifriranja koji omogućava da se operacije izvrše na šifratu, čime se dobiva šifrirani rezultat koji bi dešifriran odgovarao rezultatu operacija izvršenih na otvorenom tekstu.

Definicija 2.0.1. *Homomorfni kriptosustav je kriptosustav u kojem su skup mogućih otvorenih tekstova $\mathcal{P}$ i skup mogućih šifrata C takvi da za bilo koji $K \in \mathcal{K}$ i svaka dva šifrata $c_1 = e_K(m_1)$, $c_2 = e_K(m_2)$, vrijedi:*

- $d_K(c_1 + c_2) = m_1 + m_2$
- $d_K(c_1 \cdot c_2) = m_1 \cdot m_2$

gdje $+$ i $\cdot$ predstavljaju pripadne operacije u C i $\mathcal{P}$.

Zbog toga što su zbrajanje i množenje često operacije koje su omogućene homomorfnim kriptosustavom, simbol $\oplus$ ćemo koristiti uz šifrate za operaciju koja rezultira šifratom sume otvorenih tekstova, a simbol $\otimes$ za operaciju koja rezultira šifratom produkta otvorenih tekstova:

$$d_K(c_1 \oplus c_2) = m_1 + m_2$$
$$d_K(c_1 \otimes c_2) = m_1 \cdot m_2.$$

Najčešće, operacije $\oplus$ i $\otimes$ su implementirane polinomijalnim algoritmima.

2.1 Svojstva homomorfnih kriptosustava

Homomorfni kriptosustavi imaju neka zanimljiva matematička svojstva. Navest ćemo neka od njih.

Rešifriranje [1]

Rešfrirajući kriptosustavi su kriptosustavi sa dodatnim svojstvom da je za dani javni ključ K_e i funkciju šifriranja $E_{K_e}(m, r)$ poruke $m \in \mathcal{M}$, pomoću ključa K_e i nasumičnog broja $r \in \mathbb{Z}$, moguće učinkovito konvertirati šifrat $E_{K_e}(m, r)$ u drugi šifrat $E_{K_e}(m, r')$ koji se ni po čemu ne može razlikovati od novog šifrata poruke m pomoću ključa K_e.

Očito je da je svaki vjerojatnosni kriptosustav rešfrirajući. Bez smanjenja općenitosti, možemo pretpostaviti da je kriptosustav aditivno homomorfan. Za dani $E_{K_e}(m, r)$ i javni ključ K_e, možemo izračunati $E_{K_e}(0, r'')$ za nasumičan broj r'' i zatim izračunati sljedeće:

$$E_{K_e}(m, r) \oplus E_{K_e}(0, r'') = E_{K_e}(m + 0, r') = E_{K_e}(m, r')$$

pri čemu je r' nasumičan broj.

Slučajna samoreducibilnost [2]

Zajedno sa svojstvom rešifriranja dolazi svojstvo slučajne samoreducibilnosti s obzirom na problem dekriptiranja, tj. pogađanja otvorenog teksta iz šifrata bez poznavanja ključa. Slučajna samoreducibilnost je svojstvo da algoritam koji može dekriptirati neki netrivijalni komad šifrata može također dekriptirati cijeli šifrat sa značajnom vjerojatnošću.

Provjerljivost [3]

Ako je šifriranje provjerljivo, daje nam mogućnost da provjerimo ispravnost šifrata bez da kompromitiramo tajnost otvorenog teksta. Za primjer, ovo je korisno u elektroničkom glasovanju kako bi se uvjerio promatrač da je šifrirani naziv kandidata upravo naziv nekog od ponuđenih kandidata sa liste.

2.2 Ograničenja homomorfnih kriptosustava

Očito je da niti jedan homomorfni kriptosustav ne može biti $IND - CCA2$ siguran. To slijedi iz činjenice da je svaki homomorfni kriptosustav plastičan.

Teorem 2.2.1. *Ako je kripotustav C homomorfan, onda nije $IND - CCA2$ siguran.*

Dokaz. Promotrimo igru iz 1.1.7. U koraku 5 napadač je odabrao dvije poruke m_0 i m_1 te od izazivača dobio $e_K(m_b)$ te mora odrediti je li $b = 0$ ili $b = 1$. Napadač može tražiti

[1] Re-randomizable encryption / Re-encription
[2] Random self-reducibility
[3] Verifiable encryptions / fair encryptions

šifrat bilo koje konstante c te iskoristiti homomorfna svojstva kriptosustava da izračuna $e_K(m_b) \oplus e_K(c) = e_K(m_b + c)$. Napadač tada šalje $e_K(m_b + c)$ crnoj kutiji i dobiva dešifrirano $m_b + c$ te tada jednostavno određuje li tražena poruka m_0 ili m_1. □

Za daljnju analizu, pretpostavit ćemo da operacije $\oplus$ i $\otimes$ imamo implementirane pomoću crne kutije koja za dana dva šifrata i operaciju daje šifrat rezultata koji odgovara operaciji na otvorenim tekstovima. Iako pretpostavka možda izgleda nerealna, možemo zamisliti da je realizirana kao zaštićeni hardver koji može dešifrirati poruke, izračunati jednostavne aritmetičke operacije i zatim šifrirati rezultat.

Pokazat ćemo da ako je neki (ne-homomorfni) kriptosustav $IND-CCA1$ siguran, tada je njegova verzija sa crnom kutijom kao homomorfni kriptosustav, također $IND-CCA1$ sigurna.

Prisjetimo se igre iz 1.1.6. Neka je $\mathcal{O}^{e,d}$ crna kutija koja odgovara na upite za šifriranje i dešifriranje, te neka je $\mathcal{O}^{e}$ crna kutija koja odgovara samo na upite za šifriranje. U drugom koraku je napadaču dan pristup kutiji $\mathcal{O}^{e,d}$ a u petom koraku napadač ima pristup kutiji $\mathcal{O}^{e}$. Kako bismo pokazali da je homomorfna verzija sa crnom kutijom kriptosustava C koji je $IND-CCA1$ siguran (ali nije bio homomorfan) također $IND-CCA1$ sigurna, crne kutije u koracima dva i pet ćemo zamijeniti sa $\mathcal{O}^{e,d,\oplus,\otimes}$ i $\mathcal{O}^{e,\oplus,\otimes}$, respektivno. Tako iste crne kutije osim operacija šifriranja i/ili dešifriranja odgovaraju i na upite za homomorfne operacije.

Teorem 2.2.2. *Neka je C jedan $IND-CCA1$ siguran kriptosustav takav da su na njegovim šifratima moguće operacije $\oplus$ i $\otimes$. Tada je njegova homomorfna verzija sa crnom kutijom također $IND-CCA1$ sigurna.*

Dokaz. Teorem vrijedi ako zamjena crnih kutija $\mathcal{O}^{e,d}$ i $\mathcal{O}^{e}$ iz definicije 1.1.6 sa crnim kutijama $\mathcal{O}^{e,d,\oplus,\otimes}$ i $\mathcal{O}^{e,\oplus,\otimes}$ ne daje napadaču dovoljnu prednost da bi pobijedio u igri sa vjerojatnošću većom od $\frac{1}{2} + \nu(\epsilon)$, gdje je ϵ preddefinirani sigurnosni parametar, a funkcija ν je beznačajna.

U koraku dva, pogledajmo primjer kada je zahtjev za operacije $\oplus$ i $\otimes$ poslan crnoj kutiji $\mathcal{O}^{e,d,\oplus,\otimes}$. Isti zahtjev je mogao biti simuliran pristupom crnoj kutiji $\mathcal{O}^{e,d}$ tako da su dešifrirane obje poruke, izračunata tražena operacija te rezultat poslan sa zahtjevom za šifriranje. Tako napadač nije dobio nikakvu dodatnu prednost u koraku dva.

Dalje, promotrimo peti korak igre. Pristup crnoj kutiji $\mathcal{O}^{e}$ nije dovoljan za simulaciju crne kutije $\mathcal{O}^{e,\oplus,\otimes}$. Umjesto toga, pokazat ćemo da su informacije vraćene iz kutije $\mathcal{O}^{e}$ nerazlučive od izlaznih informacija crne kutije $\mathcal{O}^{e,\oplus,\otimes}$. Pretpostavimo da napadač zamijeni crnu kutiju $\mathcal{O}^{e,\oplus,\otimes}$ crnom kutijom $\mathcal{O}^{e,+,\times}$, koja za dani ulaz $(e_K(m_1), e_K(m_2), \{+,\times\})$ vraća uvijek $e_K(0)$. Takva crna kutija se očito može simulirati samo pristupom crnoj kutiji $\mathcal{O}^{e}$. Iz razloga što je C kriptosustav koji je $IND-CCA1$ siguran, napadač ne može razlikovati šifrate $e_K(0)$, $e_K(m_1 + m_2)$ i $e_K(m_1 m_2)$. Stoga, napadačeva prednost u razlikovanju upita kutijama $\mathcal{O}^{e,\oplus,\otimes}$ i $\mathcal{O}^{e,+,\times}$ je beznačajna pa iz toga slijedi da je prednost dobivena napadačevim pristupom crnoj kutiji $\mathcal{O}^{e,\oplus,\otimes}$ također beznačajna.

Zbog toga što napadač ne dobiva više od beznačajne prednosti u oba koraka dva i pet, napadač ne može pobijediti u igri pristupom crnim kutijama $\mathcal{O}^{e,d,\oplus,\otimes}$ i $\mathcal{O}^{e,\oplus,\otimes}$ sa vjerojatnosti većom od $\frac{1}{2}+\nu(\epsilon)$ pa je verzija sa crnom kutijom kao homomorfnim kriptosustavom također $IND-CCA1$ sigurna. □

Teorem 2.2.3. *$IND-CCA1$ je najmanja gornja ograda sigurnosti homomorfnih kriptosustava.*

Dokaz. Teorem slijedi direktno iz teorema 2.2.1 i 2.2.2. □

Iako je teorem 2.2.2 pokazao da je $IND-CCA1$ sigurnost dostižna, dokaz se oslanja na to da su homomorfne operacije realizirane kao idealan model crne kutije. U praksi, operacije $\oplus$ i $\otimes$ su polinomni algoritmi, a ne crne kutije. Je li $IND-CCA1$ sigurnost dostižna u algoritamskoj varijatnti je još uvijek otvoren problem. Nemogućnost ostvarivanja $IND-CCA2$ sigurnosti se ne oslanja na model crne kutije pa i u algoritamskoj varijanti možemo tvrditi da je $IND-CCA1$ gornja ograda, ali postoji mogućnost da to nije najmanja gornja ograda.

2.3 Primjene homomorfnih kriptosustava

Zaštita mobilnih agenata [4]

Budući da su sve uobičajene arhitekture računala bazirane na nizovima nula i jedinica i zahtijevaju samo operacije zbrajanja i množenja, homomorfna enkripcija nudi mogućnost da se kriptira cijeli program tako da ga i dalje bude moguće izvršiti. Zaštita mobilnih agenata homomorfnom enkripcijom može biti napravljena na dva načina:
(*i*) *računanje sa šifriranom funkcijom* i
(*ii*) *računanje sa šifriranim podacima*.
Računanje sa šifriranim funkcijama je poseban način zaštite mobilnih agenata. U takvom scenariju, tajna funkcija se javno izračunava na takav način da funkcija ostane tajna. Koristeći homomorfni kriptosustav tajnu funkciju možemo izračunati tako da očuvamo njezinu tajnost. Također, možemo podatke prethodno kriptirati pa koristeći svojstvo homomorfnosti računati sa šifriranim podacima i očuvati tajnost podataka. Više detalja može se pronaći u [12].

[4] Protection of mobile agents

Višekorisničko računanje [5]

U višekorisničkom računanju, konačni broj korisnika $p_1, p_2, \ldots, p_N$, od kojih svaki ima svoj privatni set podataka $d_1, d_2, \ldots, d_N$, želi zajednički izračunati vrijednost javne funkcije na svim privatnim podacima $F(d_1, d_2, \ldots, d_N)$, ali tako da sačuvaju svoj set podataka tajnim. Detaljnije se može vidjeti u [4].

Dijeljenje tajne [6]

Dijeljenje tajne se odnosi na metode za distribuciju tajne među skupinom sudionika, od kojih svatko ima samo jedan dio tajne. Tajna se može rekonstruirati samo kada dovoljno korisnika međusobno surađuje, pojedinačni dijelovi tajne su sami za sebe neupotrebljivi.

Model praga [7]

Višekorisničko računanje i dijeljenje tajne su primjeri modela praga. Potrebna je suradnja barem $t + 1$ od ukupno N korisnika da bi neka informacija koja je podijeljena između svih bila upotrebljiva. U tom slučaju t označava prag (eng. threshold).

Dokazi bez poznavanja [8]

Problem kod uporabe tajnih informacija za autentifikaciju je činjenica da ih je potrebno izreći. Prilikom svakog izricanja tajne informacije ona je izložena prisluškivanju. Tajna informacija može biti lozinka, neki odgovor na pitanje, niz slika te drugo. Dokazi bez poznavanja bi trebali jednu stranu u komunikaciji uvjeriti kako druga strana poznaje određenu tajnu informaciju bez da se ona izreče. Detaljnije se može vidjeti u [3].

Elektroničko glasovanje [9]

U elektroničkom glasovanju svojstvo homomorfnosti nam daje alat kojim možemo odrediti ishod izbora iz šifriranih glasova bez da dešifriramo pojedinačne glasove. Detaljnije u [6] te konkretan primjer u poglavlju 3.3.

[5] Multiparty computation
[6] Secret sharing scheme
[7] Threshold scheme
[8] Zero-knowledge proofs
[9] Election schemes

Vodeni žig i potpis [10]

Sustav digitalnog vodenog žiga i potpisa dodaje dodatne informacije u digitalne podatke. Svojstvo homomorfnosti se koristi da bi se dodala oznaka prethodno šifriranim podacima. Općenito, vodeni žig se koristi da bi se identificirao vlasnik/prodavač digitalne robe kako bi se osigurala autorska prava. U sustavu potpisa, osoba koja kupuje podatke mora biti prepoznatljiva trgovcu kako se podaci ne bi ilegalno distribuirali. Detaljnije u [1] i [11].

Sustav nesvjesnog prijenosa [11]

U kriptografiji, sustav nesvjesnog prijenosa je vrsta protokola u kojoj pošiljatelj prenosi jedan od potencijalno mnogo komada informacije primatelju, ali ostaje nesvjestan koji je komad informacije (ako je uopće ijedan) prenesen. Detaljnije u [10].

Sustav opredjeljenja [12]

Sustav opredjeljenja omogućuje da se jedna strana obveže na odabrane vrijednosti (ili izabranu izjavu) imajući to skriveno od drugih, s mogućnošću da se kasnije otkrije odabrana vrijednost. Sustavi opredjeljenja su dizajnirani tako da strana ne može promijeniti vrijednost ili izjavu nakon što se opredijelila za nju, tj. sustav opredjeljenja je obvezujući. Sustavi opredjeljenja imaju važnu primjenu u velikom broju kriptografskih protokola, uključujući sigurno bacanje novčića, dokaze bez poznavanja i sigurno računanje.

Sustav opredjeljenja možemo vizualizirati tako da zamislimo pošiljatelja koji stavi poruku u zaključanu kutiju i daje tu kutiju primatelju. Poruka u kutiji je skrivena od primatelja, koji sam ne može otvoriti zaključanu kutiju. Budući da primatelj ima kutiju u kojoj je poruka, ta poruka se ne može mijenjati. Može se otkriti tek ako pošiljatelj odluči primatelju dati ključ u nekom kasnijem trenutku. Detaljnije se može vidjeti u [15].

Lutrija [13]

U kriptografskoj lutriji, broj koji označava pobjedničku kartu mora biti zajednički i nasumično izabran od strane svih sudionika. To je moguće koristeći homomorfnu enkripciju, svaki sudionik odabere slučajni broj koji zatim kriptira. Zatim, zbog svojstva homomorfnosti, lako možemo izračunati šifriranu sumu odabranih vrijednosti. Kombinacija te činjenice i modela praga nas vodi do željene funkcionalnosti. Detaljnije se može vidjeti u [6].

[10] Watermarking and fingerprinting schemes
[11] Oblivious transfer
[12] Commitment scheme
[13] Lottery protocols

Anonimne mreže [14]

Sustavi anonimne mreže omogućavaju anonimnost svih pošiljatelja tako što prikupljaju šifrirane poruke od više korisnika. Možemo zamisliti sustav anonimne mreže kao sustav koji prikuplja šifrate i prosljeđuje pripadne otvorene tekstove u slučajno permutiranom redoslijedu. U takvom scenariju, privatnost je osigurana tako što zahtijevamo da je permutacija koja povezuje ulazne šifrate i izlazne otvorene tekstove poznata samo mreži. Time određivanje parova šifrata i otvorenog teksta nije efikasnije od nasumičnog pogađanja. Poželjno svojstvo u takvom sustavu je višestruka enkripcija koja je omogućena korištenjem homomorfne enkripcije. Detaljnije u [8].

Agregacija podataka unutar senzorske mreže [15]

U bežičnoj senzorskoj mreži agregacija podataka unutar mreže je način kako se može smanjiti količina podataka koja se bežičnim putem mora poslati baznoj stanici. Međutim, ta tehnika otvara pitanje privatnosti i sigurnosti podataka. U primjenama kao što su zdravstvo i vojni nadzor, agregacija mora biti obavljena tako da osjetljivi podaci ne budu otkriveni čvoru koji ih obrađuje. Homomorfna enkripcija omogućuje da svaki senzor sačuva privatnost podataka tako da ih kriptira prije slanja u mrežu i obavljanja agregacije. Detaljnije u [13].

2.4 Paillierov kriptosustav

Definicija 2.4.1. *Neka je $k \in \mathbb{Z}$ proizvoljan. Skupove $\mathbb{Z}_k$ and $\mathbb{Z}_k^*$ definiramo kao*

$$\mathbb{Z}_k = \{z \mid z \in \mathbb{Z}, 0 \leq z < k\}$$
$$\mathbb{Z}_k^* = \{z \mid z \in \mathbb{Z}, 0 \leq z < k, nzd(z,k) = 1\}$$

Propozicija 2.4.2. *Za bilo koji $x \in \mathbb{Z}_n$ vrijedi*

$$(1+n)^x = 1 + xn \bmod n^2$$

Opis Paillierovog kriptosustava

- *Generiranje ključa*

  Neka je $n = p \cdot q$, gdje su p i q prosti brojevi. Neka je $g \in \mathbb{Z}_{n^2}^*$. Javni ključ je $PK = (n, g)$, a tajni ključ je $SK = \lambda$, gdje je λ definirana kao $\lambda = nzv(p-1, q-1)$.

[14] Mix-nets

[15] Data aggregation in wireless sensor network

- *Šifriranje*

  Neka je m poruka koju želimo šifrirati, $m \in \mathbb{Z}_n$. Slučajno izaberemo $x \in \mathbb{Z}_n^*$ i izračunamo šifrat $c = g^m \cdot x^n \bmod n^2$.

- *Dešifriranje*

  Da bi dešifrirali šifrat c, računamo $m = L(c^\lambda \bmod n^2)/L(g^\lambda \bmod n^2) \bmod n$ pri čemu funkcija L kao ulaz uzima elemente iz skupa $\mathcal{S}_n = \{u < n^2 \mid u = 1 \bmod n\}$ te se izračunava kao $L(u) = \frac{u-1}{n}$.

Korektnost kriptosustava

Cijeli brojevi $c^{\lambda(n)} \bmod n^2$ i $g^{\lambda(n)} \bmod n^2$ su jednaki 1 kad se potenciraju na n što znači da su to n-ti korijeni jedinice. Nadalje, takvi korijeni su oblika $(1 + n)^\beta = 1 + \beta n \bmod n^2$. Kao posljedica toga, L-funkcija nam omogućava izračunati takve $\beta \bmod n$ i $L((g^m)^{\lambda(n)} \bmod n^2) = m \times L(g^{\lambda(n)} \bmod n^2) \bmod n$.

Homomorfna svojstva Paillierovog kriptosustava

Značajna osobina Paillierovog kriptosustava su njegova homomorfna svojstva, zajedno sa njegovim nedeterminističkim šifriranjem. Kako je funkcija šifriranja aditivno homomorfna, možemo opisati sljedeće identitete:

- *Homomorfno zbrajanje*

  Produkt dva šifrata će se dešifrirati kao zbroj odgovarajućih otvorenih tekstova,

  $$D(E(m_1, r_1) \cdot E(m_2, r_2) \bmod n^2) = m_1 + m_2 \bmod n.$$

  Produkt šifrata i otvorenog teksta kao potencije od g će se dešifrirati kao zbroj odgovarajućih otvorenih tekstova,

  $$D(E(m_1, r_1) \cdot g^{m_2} \bmod n^2) = m_1 + m_2 \bmod n.$$

- *Homomorfno množenje*

  Šifrirani otvoreni tekst na potenciju drugog (nešifriranog) otvorenog teksta će se dešifrirati kao produkt dva otvorena teksta,

  $$D(E(m_1, r_1)^{m_2} \bmod n^2) = m_1 m_2 \bmod n,$$
  $$D(E(m_2, r_2)^{m_1} \bmod n^2) = m_1 m_2 \bmod n.$$

Općenitije, šifrirani otvoreni tekst na potenciju konstante k će se dešifrirati kao produkt otvorenog teksta i konstante,

$$D(E(m_1, r_1)^k \bmod n^2) = km_1 \bmod n.$$

Međutim, ako imamo dva šifrata šifrirana Paillierovim kriptosustavom, nema poznatog načina da izračunamo njihov šifrirani produkt bez da znamo pripadni tajni ključ.

2.5 Paillierov kriptosustav s modelom praga

Označit ćemo sa $\triangle = n!$, gdje je n broj servera između kojih će biti podijeljen ključ za dešifriranje.

Algoritam

Opis Paillierovog kriptosustava s modelom praga:

- *Generiranje ključa*

  Neka je $n = p \cdot q$, gdje su p i q prosti brojevi takvi da je $p = 2p' + 1$ i $q = 2q' + 1$ te je $nzd(n, \varphi(n)) = 1$. Neka je $h = p'q'$.

  Neka je β slučajno odabrani element iz $\mathbb{Z}_n^*$, slučajno odabrani $(a, b) \in \mathbb{Z}_n^* \times \mathbb{Z}_n^*$ te neka je $g = (1 + n)^a \cdot b^n \bmod n^2$.

  Tajni ključ $SK = \beta \cdot n$ je podijeljen između n servera. Neka je $a_0 = \beta h$. Nasumično izaberemo t vrijednosti a_i iz skupa $\{0, \ldots, n \times h - 1\}$. Neka je $f(X) = \sum_{i=0}^{t} a_i X^i$. Dio ključa s_i na i-tom serveru P_i je $f(i) \bmod nh$.

  Javni ključ PK se sastoji od n, g te vrijednosti $\theta = L(g^{h\beta}) = ah\beta \bmod n$.

  Neka je $VK = v$ generator cikličke grupe $\mathbb{Z}_{n^2}^*$. Verifikacijski ključevi VK_i su dobiveni formulom $v^{\triangle s_i} \bmod n^2$.

- *Šifriranje*

  Neka je m poruka koju želimo šifrirati, $m \in \mathbb{Z}_n$. Slučajno izaberemo $x \in \mathbb{Z}_n^*$ i izračunamo šifrat $c = g^m \cdot x^n \bmod n^2$

- *Dešifriranje po dijelovima*

  i-ti server P_i računa svoj dio $c_i = c^{2\triangle s_i} \bmod n^2$ koristeći svoj dio ključa s_i. Također, i-ti server kreira dokaz bez poznavanja da je dešifriranje ispravno, kojim osigurava da su $c^{4\triangle} \bmod n^2$ i $v^{\triangle} \bmod n^2$ izračunati sa istim eksponentom s_i u postupku računanja c_i^2 i v_i.

- *Spajanje*

  Ako imamo manje od t valjanih dijelova, dešifriranje nije moguće. Inače, neka je S skup od $t+1$ valjanih dijelova pa računamo $m = L(\prod_{j\in S} c_j^{2\mu_{0,j}^S} \bmod n^2) \times \frac{1}{4\triangle^2\theta} \bmod n$ pri čemu je $\mu_{0,j}^S = \triangle \times \prod_{j'\in S\setminus\{j\}} \frac{j'}{j'-j} \in \mathbb{Z}$.

  Faktor $\triangle$ koristimo kako bi ostali u skupu cijelih brojeva te kako bi izbjegli otkrivanje tajne vrijednosti h.

2.6 Dokazi bez poznavanja [16]

Razmatrat ćemo samo interaktivne verzije protokola. Označavat ćemo sa P osobu koja nešto dokazuje (eng. Prover), a sa V osobu koja to isto želi provjeriti (eng. Verifier). Teoreme iz ovog poglavlja nećemo dokazivati, dokazi se mogu vidjeti u [2].

Dokaz o poznavanju šifrirane poruke

Neka je n k-bitni RSA modul. Za dani $c = g^m r^n \bmod n^2$ osoba P želi uvjeriti osobu V da poznaje vrijednost m bez da istu otkrije.

1. P odabire slučajni $x \in \mathbb{Z}_n$ i $s \in \mathbb{Z}_n^*$. Računa $u = g^x s^n \bmod n^2$ kojeg prosljeđuje osobi V.
2. V izabire $e \in [0, A]$ i šalje taj e osobi P.
3. P računa $v = x - em \bmod n$, $w = sr^{-e} g^{(x-em)\div n} \bmod n$ i šalje osobi V. [17]
4. V provjerava da li vrijedi $g^v c^e w^n = u \bmod n^2$.

Pretpostavimo da osoba P zna vrijednost m. Slijedeći prethodni protokol imamo

$$g^v c^e w^n = g^{x-em}(g^m r^n)^e (sr^{-e} g^{(x-em)\div n})^n = g^{x-em} g^{me} r^{ne} s^n r^{-en} g^{n((x-em)\div n)} = g^x s^n = u \bmod n^2$$

pa će osoba V sa vjerojatnošću 1 znati da osoba P poznaje vrijednost m.

[16] Zero-knowledge proofs

[17] $A \div B := \left\lfloor \frac{A}{B} \right\rfloor$

Dokaz da šifrirana poruka leži u danom skupu poruka

Neka je n k-bitni RSA modul, $S = \{m_1, \ldots, m_p\}$ javni skup od p poruka te neka je $c = g^{m_i} r^n \bmod n^2$ šifrat od poruke m_i pri čemu je i tajan. Sljedećim protokolom osoba P uvjerava osobu V da je c šifrat neke poruke iz S:

1. P odabire slučajan $\rho \in \mathbb{Z}_n^*$. Zatim nasumično odabere $p-1$ vrijednosti $\{e_j\}_{j\neq i} \in \mathbb{Z}_n$ te $p-1$ vrijednosti $\{v_j\}_{j\neq i} \in \mathbb{Z}_n^*$. Nakon toga računa $u_i = \rho^n \bmod n^2$ i $u_j = v_j^n (g^{m_j}/c)^{e_j} \bmod n^2$ za $j \neq i$. Na kraju šalje $u_{j,j\in\{1,\ldots,p\}}$ osobi V.

2. V izabire $e \in [0, A]$ i šalje taj e osobi P.

3. P računa $e_i = e - \sum_{j\neq i} e_j \bmod n$ i $v_i = \rho r^{e_i} g^{(e-\sum_{j\neq i} e_j)\div n} \bmod n$ i šalje $\{v_j, e_j\}_{j\in\{1,\ldots,p\}}$ osobi V.

4. V provjerava je li $e = \sum_j e_j \bmod n$ te vrijedi li $v_j^n = u_j (c/g^{m_j})^{e_j} \bmod n^2$ za svaki $j \in \{1, \ldots, p\}$.

U ovakvom protokolu osoba P se obvezuje na $u_1, \ldots, u_p$ kao da paralelno pokušava dokazati da je svaki c/g^{m_j} n-ti ostatak. Do kraja iskorištava mogućnost da unaprijed odabere $p-1$ vrijednosti e_j i izračuna lažne vrijednosti $u_j = v_j^n (g^{m_j}/c)^{e_j} \bmod n^2$ gdje su v_j finalni odgovori koji su slučajno odabrani iz $\mathbb{Z}_n^*$.

Dokaz jednakosti otvorenih tekstova

Neka su $n_1, \ldots, n_p$ k-bitni RSA moduli. Za danih p šifrata $c_j = g_j^m r_j^{n_j} \bmod n_j^2$ pod pretpostavkom da dešifrati od c_j leže u intervalu $[0, 2^l]$, osoba P uvjerava osobu V da su svi c_j-ovi šifrati iste poruke m.

1. P odabire slučajni $\rho \in [0, 2^k]$ i $s_j \in \mathbb{Z}_{n_j}^*$ za svaki $j \in \{1, \ldots, p\}$. Računa $u_j = g_j^\rho s_j^{n_j} \bmod n_j^2$ kojeg prosljeđuje osobi V.

2. V izabire $e \in [0, A]$ i šalje taj e osobi P.

3. P računa $z = \rho + me$ i $v_j = s_j r_j^e \bmod n_j$ te šalje z i v_j za svaki $j \in \{1, \ldots, p\}$ osobi V.

4. V provjerava je li $z \in [0, 2^k]$ i vrijedi li $g_j^z v_j^{n_j} = u_j c_j^e \bmod n^2$ za svaki $j \in \{1, \ldots, p\}$.

Za svaki $j \in \{1, \ldots, p\}$ vrijedi da je $g_j^z v_j^{n_j} = g_j^{\rho+xe} s_j^{n_j} r_j^{en_j} = u_j c_j^e \bmod n^2$ sa vjerojatnošću 1. Nadalje, zato što je $z = \rho + me$, nejednakost $z < 2^k$ vrijedi s vjerojatnošću barem $1 - 2^l A/2^k$. Stoga će osoba P uspješno proći t iteracija protokola sa vjerojatnošću $(1 - 2^{l-k}A)^t \approx 1 - 2^{l-k}At$. Iz pretpostavki vidimo da je vjerojatnost gotovo jednaka 1.

Poglavlje 3

Elektroničko glasovanje

3.1 Uvod

U ovom poglavlju bit će opisana cijela organizacija sustava. Kao osnova koristit će se Paillierov kriptosustav, odnosno njegov algoritam s modelom praga i neki dokazi bez poznavanja povezani s ovom varijantom problema diskretnih logaritama. Opisat će se cijeli protokol glasovanja od predaje glasova do izračuna rezultata. Na kraju će se dati neke ideje za poboljšanja, kako bi se osigurala anonimnost korisnika te će se razmotriti verzija sa glasanjem bez listića.

3.2 Organizacija glasovanja

Arhitektura

Za primjer je uzet sustav orijentiran na velike grupe ljudi koji se može koristiti na nacionalnim izborima. U ovom slučaju u obzir uzimamo neka organičenja koja prilikom organizacije treba pažljivo uzeti u obzir. Na primjer, sljedeći sustav prezentira "direktne" izbore u stvarnom svijetu:

1. Lokalni centri se bave identifikacijom glasača i osiguravaju da svaki glasač glasa najviše jednom. Lokalna vlast također provjerava ispravnost glasova. Ovo u praksi može obavljati i netko drugi.

2. Lokalni centri šalju svoje lokalne rezultate u regionalni centar koji prikuplja sve lokalne rezultate, provjeravaju da li su svi njihovi lokalni centri poslali ispravne informacije te računaju regionalni rezultat.

3. Svi regionalni centri šalju regionalne rezultate nacionalnom centru koji zatim računa konačan rezultat te provjerava jesu li svi regionalni centri ispravno obavili svoj zadatak.

Kandidati

U najjednostavnijem slučaju, izborima želimo odabrati jednog pobjednika između više kandidata. Kandidati mogu biti fizičke osobe, "DA/NE" odluka ili proizvoljan skup tvrdnji među kojima moramo odabrati jednu. U nastavku ćemo taj skup kandidata označavati sa $\{1, \ldots, p\}$. Skup može sadržavati "*null*" element ako želimo omogućiti glasovanje ni za koga.

Igrači

Struktura uključuje više subjekata na raznim nivoima.

- *Glasač* - Glasač je registrirana osoba kojoj je dozvoljeno predložiti jedan glas za jednog kandidata. Svi glasači su ravnopravni, tj. svaki glas ima istu težinu u konačnom odabiru.
- *Lokalna vlast* - U najmanjem području, lokalni centri prikupljaju glasove od glasača s tog područja, računaju lokalni rezultat i prosljeđuju ga na regionalni nivo.
- *Regionalna vlast* - Regionalne vlasti preuzimaju lokalne rezultate od svih lokalnih centara koji im pripadaju, računaju regionalni rezultat i prosljeđuju ga na nacionalni nivo.
- *Nacionalna vlast* - Na najvišem nivou, nacionalna vlast preuzima regionalne rezultate te javno objavljuje ukupan broj glasova za svakog kandidata.
- *Pouzdani vremenski poslužitelj* - Ovaj igrač garantira da je glasač ostavio svoj glas prije nekog određenog vremena.

Komunikacijski model

Komunikacijski model koji se koristi je javni prijenos sa memorijom koji može biti realiziran pomoću oglasne ploče. Sva komunikacija preko oglasne ploče je javna i može ju bilo tko nadzirati. Nijedan sudionik/strana ne može obrisati nijednu informaciju, ali svaki glasač može popuniti svoj dio ploče. Da bi se kontrolirala ova pravila, nužan je neki oblik kontrole pristupa.

Sigurnosni zahtjevi

Izborni protokoli zahtijevaju sljedeća svojstva:

- *Privatnost* - Privatnost glasača osigurava da glas neće saznati niti jedna t-koalicija vlasti.
- *Javna provjerljivost* - Omogućuje nam da se bilo koja strana, uključujući i nezavisne promatrače, može uvjeriti da su izbori regularni i da je rezultat točno izračunat iz ispravnih glasova.
- *Robustnost* - Robustnost protokola osigurava da sustav može tolerirati neke kompromitirane vlasti koje pokušavaju varati tijekom računanja rezultata.
- *Anonimnost* - Pojedinačni glasovi glasača su tajni.
- *Glasanje bez listića* - Ovo svojstvo osigurava da glasač ne može konstruirati listić kojim bi dokazao kako je glasao.

Ostali napadi na sustav glasovanja

Ovaj sustav glasovanja uzima u obzir dva različita modela napada koji se mogu dogoditi:

- *Napad posrednika* - Ovaj napad uključuje posredničku vlast koja bi pokušala manipulirati finalni rezultat i takvog proslijediti na viši nivo.
- *Iznenadni napad* - U vrijeme pred sam kraj glasovanja, lokalne vlasti objavljuju trenutni lokalni rezultat. Nakon toga glasači koji su čekali objavu tih rezultata, a nisu zadovoljni ishodom, odjednom žele doći glasati samo kako bi promijenili trenutni rezultat.

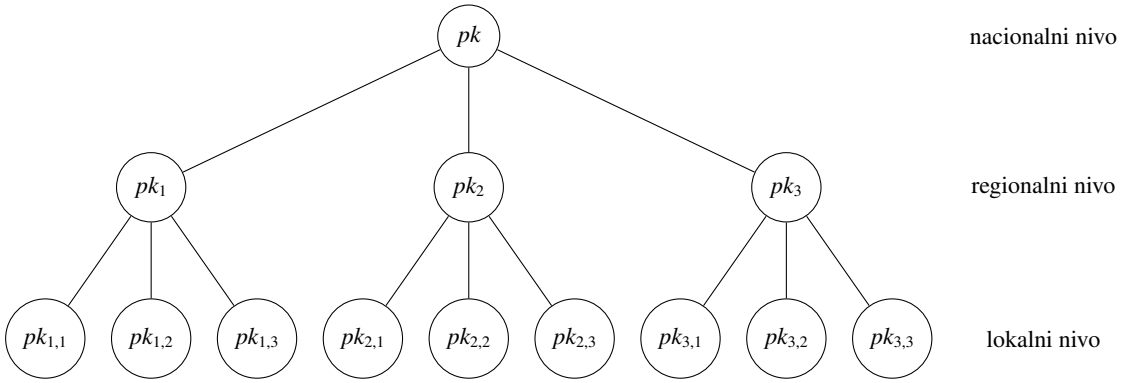
3.3 Protokol glasovanja

Opisat ćemo kompletan protokol glasovanja, koristeći kriptografske alate navedene u 2.4. Posebno želimo naglasiti kreiranje glasa te komunikaciju između tri hijerarhijska nivoa vlasti: nacionalne, regionalne i lokalne.

Organizacija

Razmatramo $1 - od - p$ izbore gdje biramo jednog kandidata od prijavljenih p. U inicijalizacijskom koraku, svaka vlast, na svakom nivou, generira svoj javni ključ i certificira ga

sa nezavisnom upravom za certifikaciju. Koristit ćemo sljedeću hijerarhijsku notaciju: pk za nacionalnu vlast, pk_i za regionalnu vlast i $pk_{i,j}$ za lokalnu vlast.



Slika 3.1: Organizacija vlasti

Nakon početnog koraka, svaka vlast objavljuje na svojoj oglasnoj ploči točne javne ključeve. Na primjer, i-ta regionalna vlast objavljuje ključeve pk, pk_i, a j-ta lokalna vlast unutar i-te regije objavljuje ključeve pk, pk_i, $pk_{i,j}$.

Označimo sa l ukupan broj glasača i postavimo cijeli broj M tako da bude veći od l. Možemo recimo izabrati $M = 2^{\lceil \log_2 l \rceil}$, potenciju broja 2 neposredno veću od l.

Glasovanje

Uzmimo za primjer glasača sa javnim ključevima pk, pk_i, $pk_{i,j}$ koji želi glasati za m-tog kandidata. On glasa na sljedeći način:

1. Sa oglasne ploče preuzme 3 javna ključa za svoje područje pk, pk_i, $pk_{i,j}$.

2. Koristeći svaki ključ, on šifrira broj M^m koristeći Pallierov kriptosustav i time kreira šifrate C_n, C_r i C_l redom sa nacionalnim, regionalnim i lokalnim javnim ključem.

3. Kreira 3 dokaza da uvjeri sve koji žele provjeriti da je svaki od tri šifrata ispravan glas, tj. da su to šifrati od M^m gdje je $m \in 1, \ldots, p$. Tu koristimo dokaz da je poruka unutar dozvoljenog skupa.

4. Kreira još jedan dokaz da sva tri šifrata sadrže isti glas. To se dobiva koristeći dokaz jednakosti otvorenih tekstova.

Izračun rezultata

U ovom poglavlju prikazat ćemo izračun rezultata. Prvo ćemo opisati oglasnu ploču lokalne vlasti kojoj mogu pristupiti svi glasači koji joj pripadaju. Na primjer, kroz web stranicu lokalne vlasti $A_{i,j}$.

Ime	C_l	C_r	C_n
Glasač 1	$g_{i,j}^{v_{i,j,1}}$	$g_i^{v_{i,j,1}}$	$g^{v_{i,j,1}}$
Glasač 2	$g_{i,j}^{v_{i,j,2}}$	$g_i^{v_{i,j,2}}$	$g^{v_{i,j,2}}$
$\vdots$			
Glasač k	$g_{i,j}^{v_{i,j,k}}$	$g_i^{v_{i,j,k}}$	$g^{v_{i,j,k}}$
$\vdots$			
Glasač l	$g_{i,j}^{v_{i,j,l}}$	$g_i^{v_{i,j,l}}$	$g^{v_{i,j,l}}$
Suma od $A_{i,j}$	$\sum_k v_{i,j,k}$	$\prod_k g_i^{v_{i,j,k}}$	$\prod_k g^{v_{i,j,k}}$

Tablica 3.1: Oglasna ploča lokalne vlasti

Tablica 3.1 prikazuje oglasnu ploču na koju glasač k može čitati i pisati u svoj redak, a sve ostale retke može samo pročitati.

Glasač upiše svoje ime zajedno sa svojim certifikatima, tri glasa i dokaza. Zatim potpiše sve zapise u svojem retku i na kraju vremenski poslužitelj označi glas s vremenom kad je predan.

Kad glasovanje završi, lokalna vlast mora provjeriti sve dokaze i potpise te potpise vremenskog poslužitelja. Nakon toga, lokalna vlast $A_{i,j}$ računa produkt svih ispravnih glasova u stupcima $2-4$. Dešifrira se izračunata suma prvog stupca koji odgovara njihovom javnom ključu (kao u Paillierovom kriptosustavu s modelom praga). Dešifriranje mogu provjeriti vanjski promatrači budući da je proces dešifriranja javno provjerljiv.

Lokalni rezultati se upisuju u regionalnu oglasnu ploču A_i u prvi stupac, a u stupce 2 i 3 se upisuje produkt elemenata iz svoje oglasne ploče. Razne lokalne vlasti $A_{i,j}$ koje pripadaju pod regionalnu vlast A_i imaju jednaka prava kao korisnici u prethodnoj oglasnoj ploči. Stupac C_l je zamijenjen sa dešifriranim rezultatom.

Regionalne vlasti A_i računaju sumu prvog stupca, produkte stupaca 2 i 3 i dešifriraju finalni produkt stupca 2 na isti način kao i lokalne vlasti. Na kraju provjeravaju podudaraju li se rezultati izračunati na oba načina. U oglasnu ploču nacionalne vlasti upisuju tu sumu u prvi stupac, a u drugi stupac produkt svojih lokalnih vlasti.

Lokalna vlast	$\mathcal{V}_{j,k}$	$\prod_{j,k} C_r$	$\prod_{j,k} C_n$
Lokalna vlast 1	$\sum_k v_{i,1,k}$	$\prod_k g_i^{v_{i,1,k}}$	$\prod_k g^{v_{i,1,k}}$
Lokalna vlast 2	$\sum_k v_{i,2,k}$	$\prod_k g_i^{v_{i,2,k}}$	$\prod_k g^{v_{i,2,k}}$
$\vdots$			
Lokalna vlast j	$\sum_k v_{i,j,k}$	$\prod_k g_i^{v_{i,j,k}}$	$\prod_k g^{v_{i,j,k}}$
$\vdots$			
Lokalna vlast l	$\sum_k v_{i,l,k}$	$\prod_k g_i^{v_{i,l,k}}$	$\prod_k g^{v_{i,l,k}}$
Suma od A_i	$\sum_{j,k} v_{i,j,k}$	$\prod_{j,k} g_i^{v_{i,j,k}}$ $\Downarrow$ dešifriranje $\sum_{j,k} v_{i,j,k}$	$\prod_{j,k} g^{v_{i,j,k}}$

Tablica 3.2: Oglasna ploča regionalne vlasti

Nacionalna vlast sad može izračunati produkt elemenata u drugom stupcu. Na kraju dešifrira tu sumu i provjerava odgovara li ona sumi elemenata prvog stupca.

Regionalna vlast	$\mathcal{V}_{i,j,k}$	$\prod_{i,j,k} C_n$
Regionalna vlast 1	$\sum_{j,k} v_{1,j,k}$	$\prod_{j,k} g^{v_{1,j,k}}$
Regionalna vlast 2	$\sum_{j,k} v_{2,j,k}$	$\prod_{j,k} g^{v_{2,j,k}}$
$\vdots$		
Regionalna vlast i	$\sum_{j,k} v_{i,j,k}$	$\prod_{j,k} g^{v_{i,j,k}}$
$\vdots$		
Regionalna vlast l	$\sum_{j,k} v_{l,j,k}$	$\prod_{j,k} g^{v_{l,j,k}}$
Suma od A_i	$\sum_{i,j,k} v_{i,j,k}$	$\prod_{i,j,k} g^{v_{i,j,k}}$ $\Downarrow$ dešifriranje $\sum_{i,j,k} v_{i,j,k}$

Tablica 3.3: Oglasna ploča nacionalne vlasti

Prijevare tokom izračuna

Hijerarhijska podjela pruža efikasan način distribucije javne verifikacije rezultata. Svaki glasač je u mogućnosti provjeriti je li njegov glas uzet u obzir prilikom izračuna lokalnih rezultata. Zatim, može rekurzivno provesti sličnu provjeru kojom bi se uvjerio da je sljedeća razina vlasti pravilno uzela u obzir lokalne rezultate. Na kraju je uvjeren da je njegov glas dio finalnog rezultata izbora. Pod pretpostavkom da je broj hijerarhijskih razina

jednak d (3 u našem primjeru) i ukupan broj glasača je jednak l, tada je složenost provjere $O(d^{1/l})$ što se lako izvrši na osobnom računalu.

Dodatno, ako se pokaže neka greška, neispravan skup podataka se može izuzeti iz finalnog rezultata do njihove rekalkulacije.

3.4 Moguća poboljšanja

Poslužitelj za vrijeme

Glasovanje završava u unaprijed poznato vrijeme T. Stoga, glasovi koji su nastali nakon vremena T neće biti uzeti u obzir kod računanja rezultata. U tu svrhu moramo uvesti vremenski poslužitelj koji će garantirati da su svi prikupljeni glasovi stigli prije vremena T.

Anonimnost

U stvarnom glasovanju, sa glasačkim listićima, poznat je popis aktivnih glasača, barem vlastima ako ne i ostalim glasačima. Glavni razlog tome je da se osigura da nitko ne može glasati više od jednom.

Nakon što dokaže svoj identitet, glasač kreira par tajnog i javnog ključa. Zatim, uz pomoć vlasti, dobiva certifikat za njih, koristeći slijepi potpis. Naravno, vlast mora prihvatiti samo jednu interakciju sa glasačem. Nakon toga svaki glasač ima samo jedan ovlašteni javni ključ koji zatim može koristiti kao pseudonim. Također, želimo onemogućiti da neki glasač u dogovoru sa nekom jedinicom vlasti dobije više od jednog ovlaštenog javnog ključa, odnosno više pseudonima i tako mogućnost da glasa više od jednom. Moramo koristiti distribuirani sustav slijepog potpisa tako da su potrebne sve jedinice vlasti zajedno da glasač dobije certificirani pseudonim. Tako neka jedinica vlasti sama ne može omogućiti jednom glasaču dobivanje više od jednog certificiranog pseudonima.

Distribuirana certifikacija pseudonima

Neka je $Z \in \mathbb{Z}_N^*$ javni ključ vlasti zajedno sa dovoljno velikim javnim eksponentom e, dok su distribuirani tajni ključevi parovi x_i, y_i takvi da je $x_i \in \mathbb{Z}_N^*$, a $y_i \in [0, e]$ takvi da je $Z = \prod_i x_i^e a^{y_i} \bmod N$. Da bi dobio certifikat za svoj pseudonim V, glasač nakon što potvrdi svoj identitet mora komunicirati sa svim vlastima:

1. Svaka vlast izabire slučajan $u_i \in \mathbb{Z}_N^*$ i $v_i \in [0, e]$ te računa $w_i = u_i^e a^{v_i} \bmod N$ kojeg šalje glasaču.

2. Glasač izabire nasumične $\beta \in \mathbb{Z}_N^*$ i $\alpha, \gamma \in [0, e]$ te izračunava $w = (\prod_i w_i) a^\alpha \beta^e Z^\gamma \bmod N$.

3. Glasač zatim računa $\epsilon = H(w, V)$, $c = \epsilon + \gamma \bmod e$ i šalje svim vlastima.

4. Svaka vlast računa $r_i = v_i + cy_i \bmod e$, $s_i = (v_i + cy_i) \div e$ te $t_i = u_i x_i^c a^{s_i} \bmod N$. Vlasti šalju (r_i, t_i) glasaču.

5. Glasač računa $r = \sum_i r_i + \alpha \bmod e$, $s = (\sum_i r_i + \alpha) \div e$, $s' = (c - \epsilon) \div e$ i $t = (\prod_i t_i) \beta a^s Z^{-s'} \bmod N$

Uz takvu konstrukciju imamo:

$$\begin{aligned}
a^{r_i} t_i^e &= a^{r_i} u_i x_i^c a^{s_i} = a^{r_i + e s_i} u_i^e x_i^{ce} \\
&= a^{v_i + cy_i} u_i^e x_i^{ce} = a^{v_i} u_i^e (a^{y_i} x_i^e)^c \\
&= w_i Z_i^c \bmod N \\
a^r t^e &= a^r ((\prod_i t_i) \beta a^s Z^{-s'})^e = a^r (\prod_i t_i^e) \beta^e a^{es} Z^{-es'} \\
&= a^{r+es} Z^{-es'} (\prod_i t_i^e) \beta^e = a^{\sum_i r_i + \alpha} Z^{-es'} (\prod_i t_i^e) \beta^e \\
&= (\prod_i a^{r_i} t_i^e) a^\alpha \beta^e Z^{-es'} = Z^{c-es'} (\prod_i w_i) a^\alpha \beta^e \\
&= Z^{c-es'-\gamma} w = Z^{c-c+\epsilon} w \\
&= Z^\epsilon w \bmod N.
\end{aligned}$$

gdje je $\epsilon = H(w, V)$, a V je glasačev pseudonim. Tako glasač ne može dobiti dva pseudonima s kojima bi glasao dvaput, osim ako sve vlasti ne surađuju. Razlog tome je što su za dobivanje certifikata potrebne sve vlasti jer svaka vlast ima samo dio ključa.

Radi pojednostavljenja, umjesto prvotnog potvrđivanja identiteta, svaki glasač bi mogao samo potpisati c kojeg šalje vlastima. Također, umjesto da se ključ podijeli između svih vlasti, mogli bi koristiti slijepi potpis sa modelom praga pa bi bilo dovoljna $t + 1$ jedinica vlasti za dobivanje certifikata.

Glasanje bez listića

Glasanje bez listića osigurava da glasač ne može dati nikakav dokaz kojim bi otkrio sadržaj svog glasa, čak i kad bi to želio. Cilj toga je da se onemogući da glasači prodaju svoje glasove.

Glasači bi tražili nezavisnu osobu da randomizira njihove glasove, bez modificiranja sadržaja. Glasačima neće biti dovoljna samo nova enkripcija njihovih glasova, htjet će i sljedeće:

- Imati dokaz da novi šifrat predstavlja šifrat iste originalne poruke - glasa.

  Taj dokaz naravno ne smije biti prenosiv jer bi u tom slučaju predstavljao dokaz kojim bi se mogao otkriti sadržaj originalnog šifrata glasa. Također, taj dokaz mora biti dokaz bez otkrivanja jer ne smije sadržavati informaciju o novoj slučajnoj vrijednosti korištenoj u šifriranju glasa.

- Imati dokaz da je novi šifrat validan.

  Kako glasač ne može kreirati takav dokaz sam bez da zna slučajnu vrijednost koju je uveo nezavisni randomizator, morat će takav dokaz kreirati u suradnji sa randomizatorom, ali opet bez da postoji mogućnost da u nekom trenutku neki dio iskoristi za kreiranje potvrde o sadržaju glasa.

- Kako u našem protokolu glasovanja imamo tri razine (a može ih biti i više), nezavisna osoba mora randomizirati tri glasa i glasač mora imati potvrdu da ta tri šifrata označavaju isti otvoreni tekst. Taj dokaz možemo smatrati modifikacijom dokaza jednakosti dva otvorena teksta.

Glasanje bez listića sa nezavisnim randomizatorom

U prvom koraku randomizator mora dokazati glasaču da su novi i originalni šifrati glasa ekvivalentni. Glasač je koristio r da bi šifrirao glas $c = g^{m_i} r^N \bmod N^2$ kojeg je poslao randomizatoru. Nakon toga je randomizator uveo novu slučajnu vrijednost s da bi transformirao šifrirani glas c u $c' = cs^N = g^{m_i}(rs)^N \bmod N^2$.

1. Randomizator odabire nasumičnu vrijednost $x \in \mathbb{Z}_N$ te računa $u = x^N \bmod N^2$ koju šalje glasaču.

2. Glasač odabire $e \in [0, A]$ kojeg šalje randomizatoru.

3. Randomizator s time računa $v = xs^{-e} \bmod N^2$ i šalje glasaču.

4. Na kraju glasač provjerava vrijedi li $v^N (c'/c)^e = u \bmod N^2$.

Zahvaljujući dokazu bez poznavanja s fiksnim parametrom A, sam dokaz nije prenosiv.

Drugi korak je mala preinaka dokaza da je šifrat nastao od poruke koja leži u danom skupu poruka. Koristimo istu notaciju, a randomizatora možemo gledati kao osobu koja provjerava (eng. Verifier).

1. Randomizator od glasača dobiva listu $u_{j, j \in \{1,\ldots,p\}}$. Zatim bira sljedeće faktore:

   - slučajan $\beta \in \mathbb{Z}_N$

- p vrijednosti $\{\beta_j\}_{j\in\{1,\dots,p\}} \in \mathbb{Z}_N$ takvih da je $\sum_j \beta_j = \beta \bmod N$
- p vrijednosti $\{\alpha_j\}_{j\in\{1,\dots,p\}} \in \mathbb{Z}_N^*$

Zatim računa

$$u'_j = u_j \alpha_j^N (c/g^{m_j})^{\beta_j} \bmod N^2.$$

Na kraju dobiva $e' = H(u'_1, \dots, u'_p)$ i šalje $e = e' + \beta \bmod N$ glasaču.

2. Glasač šalje skup $\{v_j, e_j\}_{j\in\{1,\dots,p\}}$ randomizatoru tako da vrijedi

$$e = \sum_j e_j \bmod N \text{ i vrijedi } v_j^N = u_j (c/g^{m_j})^{e_j} \bmod N^2$$

za svaki $j \in \{1, \dots, p\}$.

3. Randomizator računa $e'_j = e_j - \beta_j \bmod N$ i $v'_j = v_j \alpha_j s^{e'_j} \bmod N^2$ i šalje skup $\{v'_j, e'_j\}_{j\in\{1,\dots,p\}}$ glasaču.

Lako se vidi da vrijedi:

$$\begin{aligned} e' &= e - \beta = \sum_j e_j - \sum_j \beta_j = \sum_j (e_j - \beta_j) \\ &= \sum_j e'_j \bmod N, \\ v'^N_j &= v_j^N \alpha_j^N s^{Ne'_j} = u_j (c/g^{m_j})^{e_j} \alpha_j^N s^{Ne'_j} \\ &= u_j (cs^N/g^{m_j})^{e'_j} (c/g^{m_j})^{\beta_j} \alpha_j^N \\ &= u'_j (cs^N/g^{m_j})^{e'_j} \\ &= u'_j (c'/g^{m_j})^{e'_j} \bmod N^2 \end{aligned}$$

za svaki $j \in \{1, \dots, p\}$, gdje je $e' = H(u'_1, \dots, u'_p)$. Stoga je skup $\{u'_j, v'_j, e'_j\}_{j\in\{1,\dots,p\}}$ dokaz ispravnosti šifrata c'.

Bibliografija

[1] André Adelsbach, Stefan Katzenbeisser i Ahmad Reza Sadeghi, *Cryptography meets watermarking: Detecting watermarks with minimal or zero knowledge disclosure*, Signal Processing Conference, 2002 11th European, IEEE, 2002, str. 1–4.

[2] Olivier Baudron, Pierre Alain Fouque, David Pointcheval, Jacques Stern i Guillaume Poupard, *Practical multi-candidate election system*, Proceedings of the twentieth annual ACM symposium on Principles of distributed computing, ACM, 2001, str. 274–283.

[3] Ronald Cramer i Ivan Damgård, *Zero-knowledge proofs for finite field arithmetic, or: Can zero-knowledge be for free?*, Annual International Cryptology Conference, Springer, 1998, str. 424–441.

[4] Ivan Damgård, Valerio Pastro, Nigel Smart i Sarah Zakarias, *Multiparty computation from somewhat homomorphic encryption*, Advances in Cryptology–CRYPTO 2012, Springer, 2012, str. 643–662.

[5] Andrej Dujella i Marcel Maretić, *Kriptografija*, Element, 2007.

[6] Pierre Alain Fouque, Guillaume Poupard i Jacques Stern, *Sharing Decryption in the Context of Voting or Lotteries*, str. 90–104, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001, ISBN 978-3-540-45472-4, `http://dx.doi.org/10.1007/3-540-45472-1_7`.

[7] Marc Girault i Jacques Stern, *On the length of cryptographic hash-values used in identification schemes*, Annual International Cryptology Conference, Springer, 1994, str. 202–215.

[8] Philippe Golle, Markus Jakobsson, Ari Juels i Paul Syverson, *Universal re-encryption for mixnets*, Cryptographers' Track at the RSA Conference, Springer, 2004, str. 163–178.

[9] Kevin Henry, *The theory and applications of homomorphic cryptography*, (2008).

[10] Helger Lipmaa, *Verifiable Homomorphic Oblivious Transfer and Private Equality Test*, str. 416–433, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, ISBN 978-3-540-40061-5, http://dx.doi.org/10.1007/978-3-540-40061-5_27.

[11] Birgit Pfitzmann i Michael Waidner, *Anonymous fingerprinting*, International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 1997, str. 88–102.

[12] Tomas Sander i Christian F Tschudin, *Protecting mobile agents against malicious hosts*, Mobile agents and security, Springer, 1998, str. 44–60.

[13] Jaydip Sen, *Cryptography and Security in Computing*, 2012.

[14] ———, *Homomorphic Encryption: Theory & Applications*, CoRR **abs/1305.5886** (2013), http://arxiv.org/abs/1305.5886.

[15] Nigel Paul Smart, *Cryptography: an introduction*, sv. 5, McGraw-Hill New York, 2003.

Sažetak

U ovom radu obrađena je tema homomorfne enkripcije i njezina primjena u elektroničkom glasovanju. Ovaj oblik šifriranja omogućava izvršavanje različitih algebarskih operacija nad šifriranim podacima, koje su ekvivalentne istim ili različitim algebarskim operacijama nad otvorenim tekstom, bez potrebe da se podaci prije toga dešifriraju. Zbog toga je homomorfna enkripcija zanimljiva u raznim protokolima u kojima je nužno očuvanje privatnosti. Elektroničko glasovanje je jedno od najčešćih područja u kojima se primjenjuje homomorfna enkripcija. U radu je opisan cijeli protokol glasovanja od predaje glasova do izračuna rezultata. Na primjeru je pokazano na koji način izborna administracija može prebrojati glasove i objaviti konačne rezultate izbora, bez da se istovremeno dešifriraju pojedinačni glasovi. Na kraju rada dani su prijedlozi za moguća poboljšanja elektroničkog glasovanja.

Summary

This paper deals with the topic of homomorphic encryption and its application in electronic voting. This form of encryption enables the execution of various algebraic operations over the encrypted data, which are equivalent to the same or different algebraic operations over the open text without the need for the data to be previously decrypted. Because of this, homomorphic encryption is interesting in various protocols where there is a necessity to preserve privacy. Electronic voting is one of the most common areas where homomorphic encryption is applied. This paper describes the entire voting protocol from the casting of the votes to the tallying of the results. The example shows how electoral administration can count votes and announce final election results without having to simultaneously decrypt individual votes. At the end of the paper, proposals for possible improvements to electronic voting were given.

Životopis

Rođen sam 04. prosinca 1987. godine u Zagrebu. Nakon završetka srednjoškolskog obrazovanja u Gimnaziji Velika Gorica (prirodoslovno - matematički smjer), 2006. godine upisao sam Preddiplomski sveučilišni studij Matematika na Prirodoslovno - matematičkom fakultetu u Zagrebu, a 2012. godine Preddiplomski sveučilišni studij Matematika (nastavnički smjer). Završetkom tog studija i stjecanjem diplome sveučilišnog prvostupnika edukacije matematike 2014. godine, upisao sam Diplomski sveučilišni studij Računarstvo i matematika na istom fakultetu. Od 2012. do 2014. godine radio sam na poziciji administratora na odjelu teleprodaje u Hrvatskom Telekomu, a od 2014. godine radim u Odjelu koordinacije i upravljanja kanalima za poslovne korisnike u istoj kompaniji.